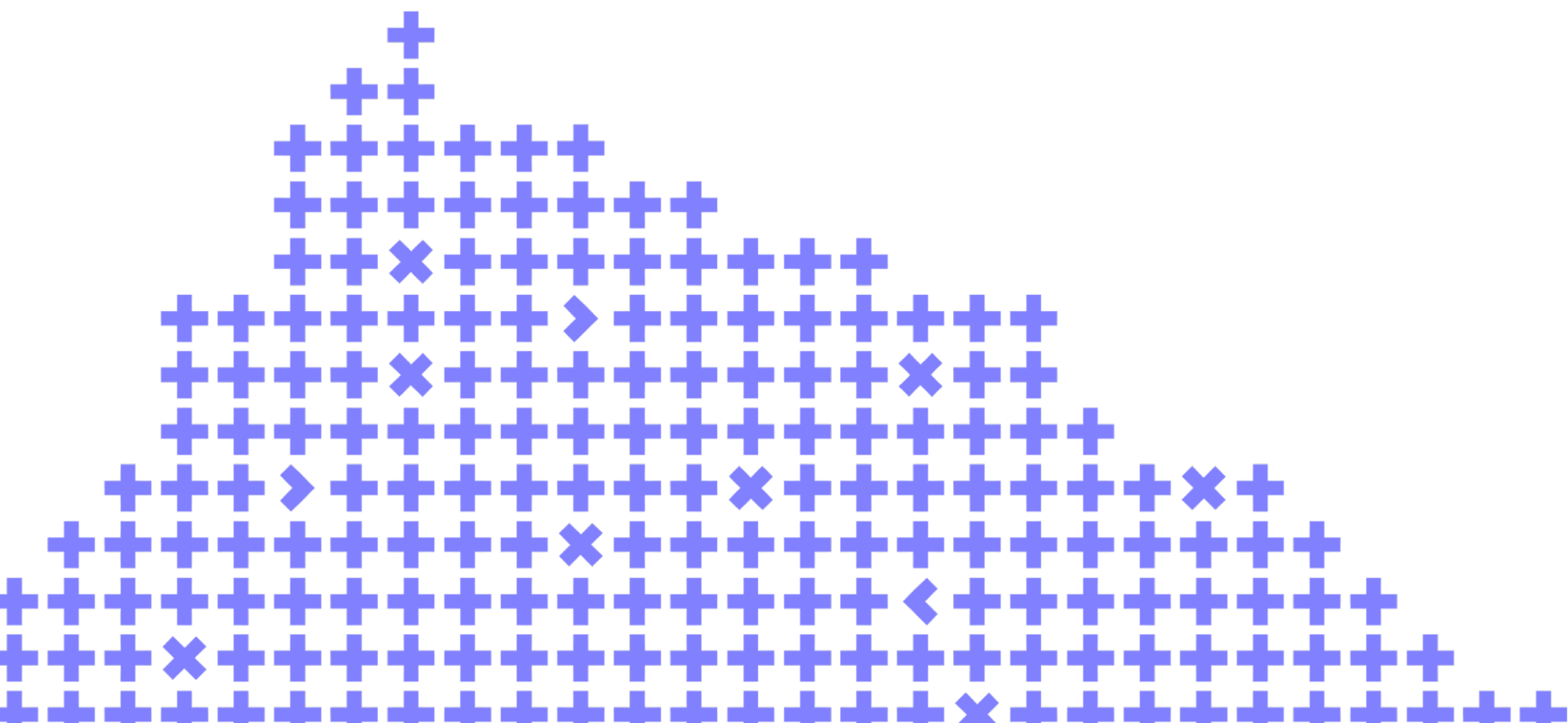


Building an open-source MongoDB-compatible database on top of PostgreSQL

Alexey Palazhchenko
FerretDB

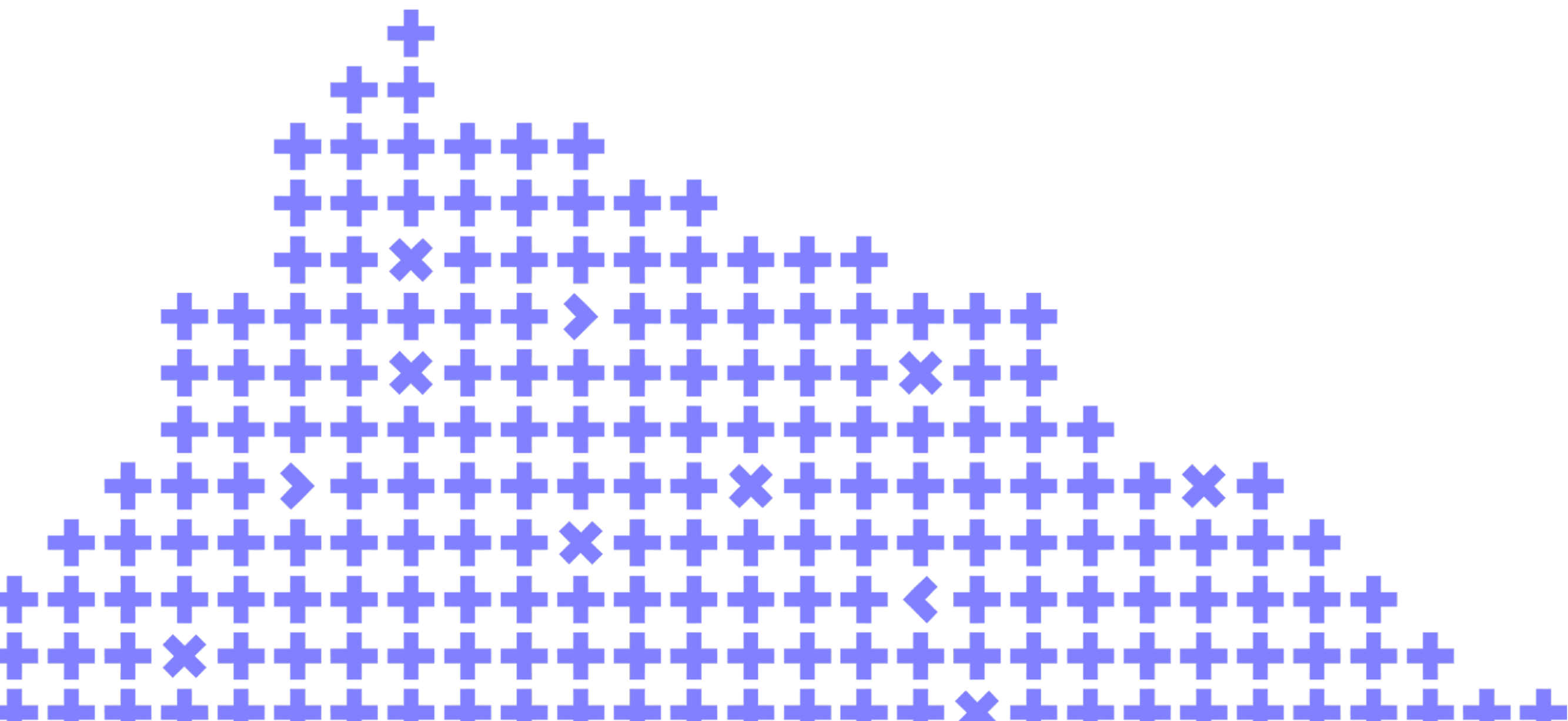


Co-organizer

Yandex

Building³ an open-source² MongoDB¹-compatible database on top of PostgreSQL⁴

Alexey Palazhchenko
FerretDB



Co-organizer

Yandex

MongoDB: The Good Parts

MongoDB: The Good Parts

- Great developer experience

MongoDB: The Good Parts

- Great developer experience
- Good drivers

MongoDB: The Good Parts

- Great developer experience
- Good drivers
- Great community (back in the day)

MongoDB: The Bad Parts

MongoDB: The Bad Parts

- Reliability, performance, scalability

MongoDB: The Bad Parts

- Reliability, performance, scalability



MongoDB: The Bad Parts

- Reliability, performance, scalability
- Does too much?



MongoDB: The Bad Parts

- Reliability, performance, scalability
- Does too much?
- Company loses money



MongoDB: The Bad Parts

- Reliability, performance, scalability
- Does too much?
- Company loses money
- All-in on Atlas



MongoDB: The Bad Parts

- Reliability, performance, scalability
- Does too much?
- Company loses money
- All-in on Atlas
- License change



Not a legal advice

Not a legal advice



imgflip.com

AGPL v3

AGPL v3

- «If you modify the Program»

AGPL v3

- «If you modify the Program»
- «All users interacting with it remotely through a computer network»

AGPL v3

- «If you modify the Program»
- «All users interacting with it remotely through a computer network»
- «Corresponding Source»

AGPL v3

- «If you modify the Program»
- «All users interacting with it remotely through a computer network»
- «Corresponding Source»
- «Including scripts to control those activities»

SSPL

SSPL

- «If you make the functionality of the Program»

SSPL

- «If you make the functionality of the Program»
- «Available to third parties as a service»

SSPL

- «If you make the functionality of the Program»
- «Available to third parties as a service»
- «Includes, without limitation, [...]»

SSPL

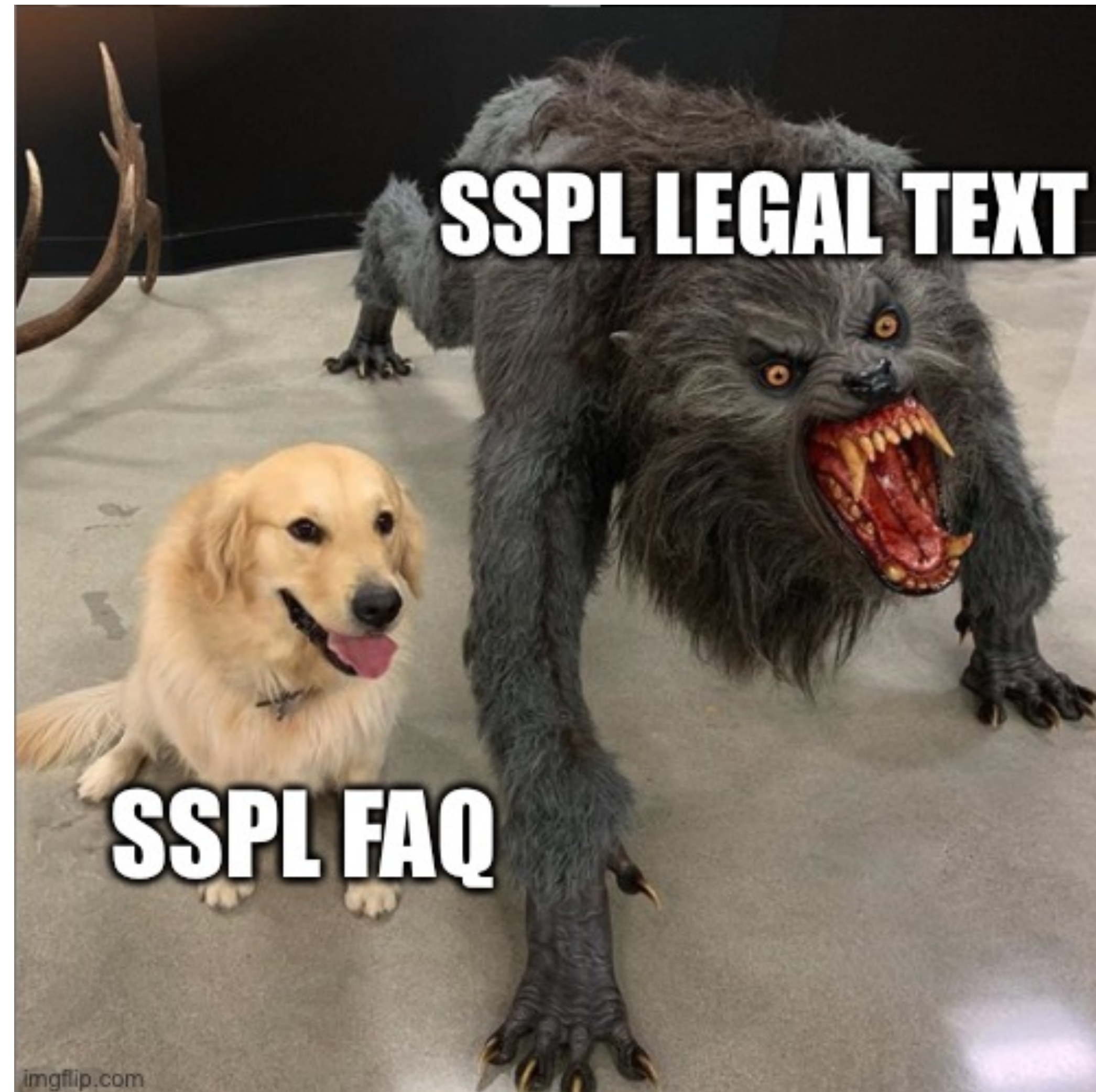
- «If you make the functionality of the Program»
- «Available to third parties as a service»
- «Includes, without limitation, [...]»
- «Service Source Code»

SSPL

- «If you make the functionality of the Program»
- «Available to third parties as a service»
- «Includes, without limitation, [...]»
- «Service Source Code»
- «Including, without limitation, management software, user interfaces, application program interfaces, [...], all such that a user could run an instance of the service»

SSPL FAQ

SSPL FAQ





**open source
initiative®**

[https://opensource.org/
sspl-not-open-source](https://opensource.org/sspl-not-open-source)

Protocol specification license

```
15 + .. note::
16 +
17 + This MongoDB Wire Protocol Specification is licensed under a
18 + `Creative Commons Attribution-NonCommercial-ShareAlike 3.0 United States License
19 + <https://creativecommons.org/licenses/by-nc-sa/3.0/us/>`__. You may
20 + not use or adapt this material for any commercial purpose, such as
21 + to create a commercial database or database-as-a-service offering.
22 +
```

Alternatives?

Alternatives?



Amazon
DocumentDB

Alternatives?



Amazon
DocumentDB



Azure
Cosmos DB

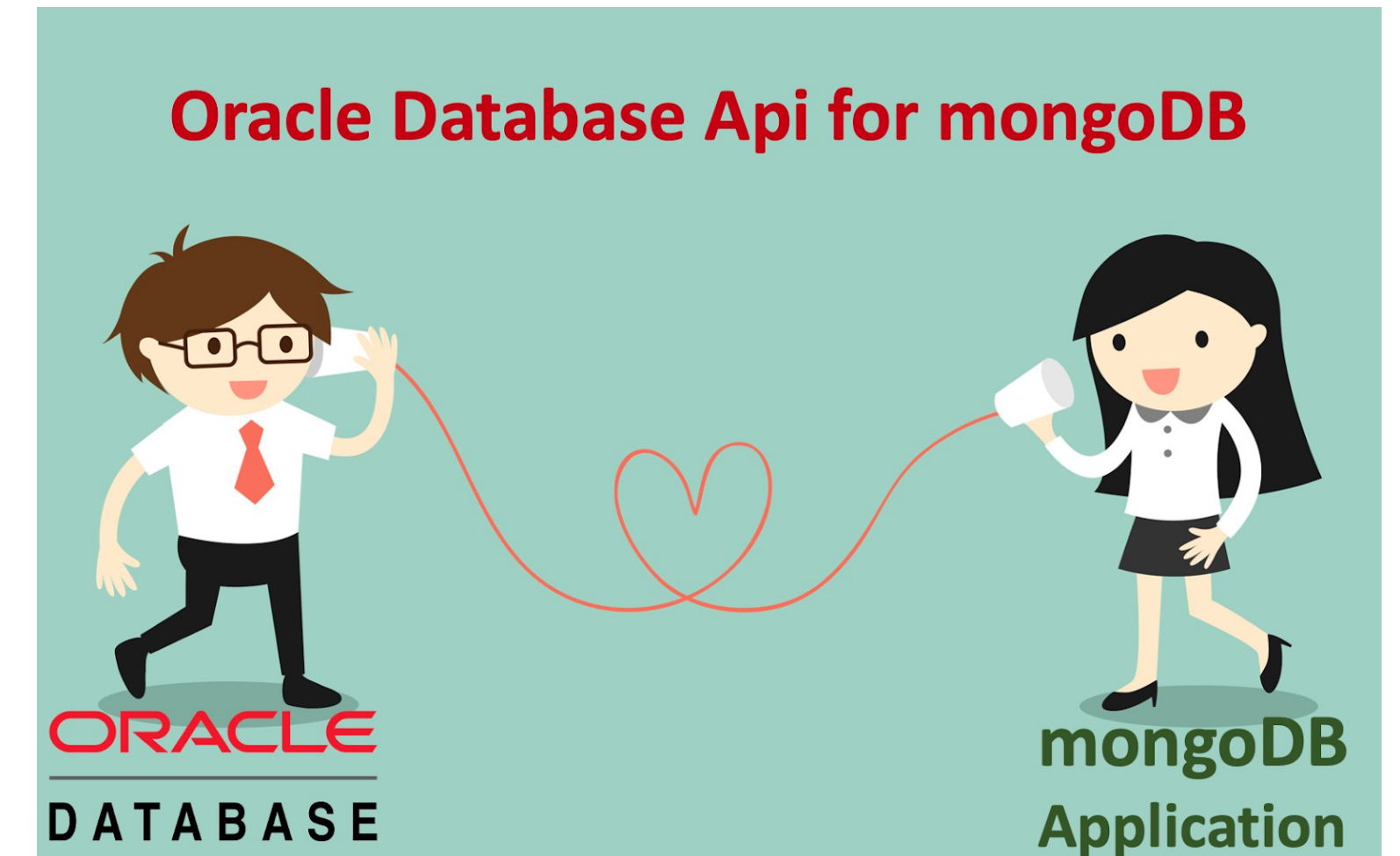
Alternatives?



**Amazon
DocumentDB**



**Azure
Cosmos DB**



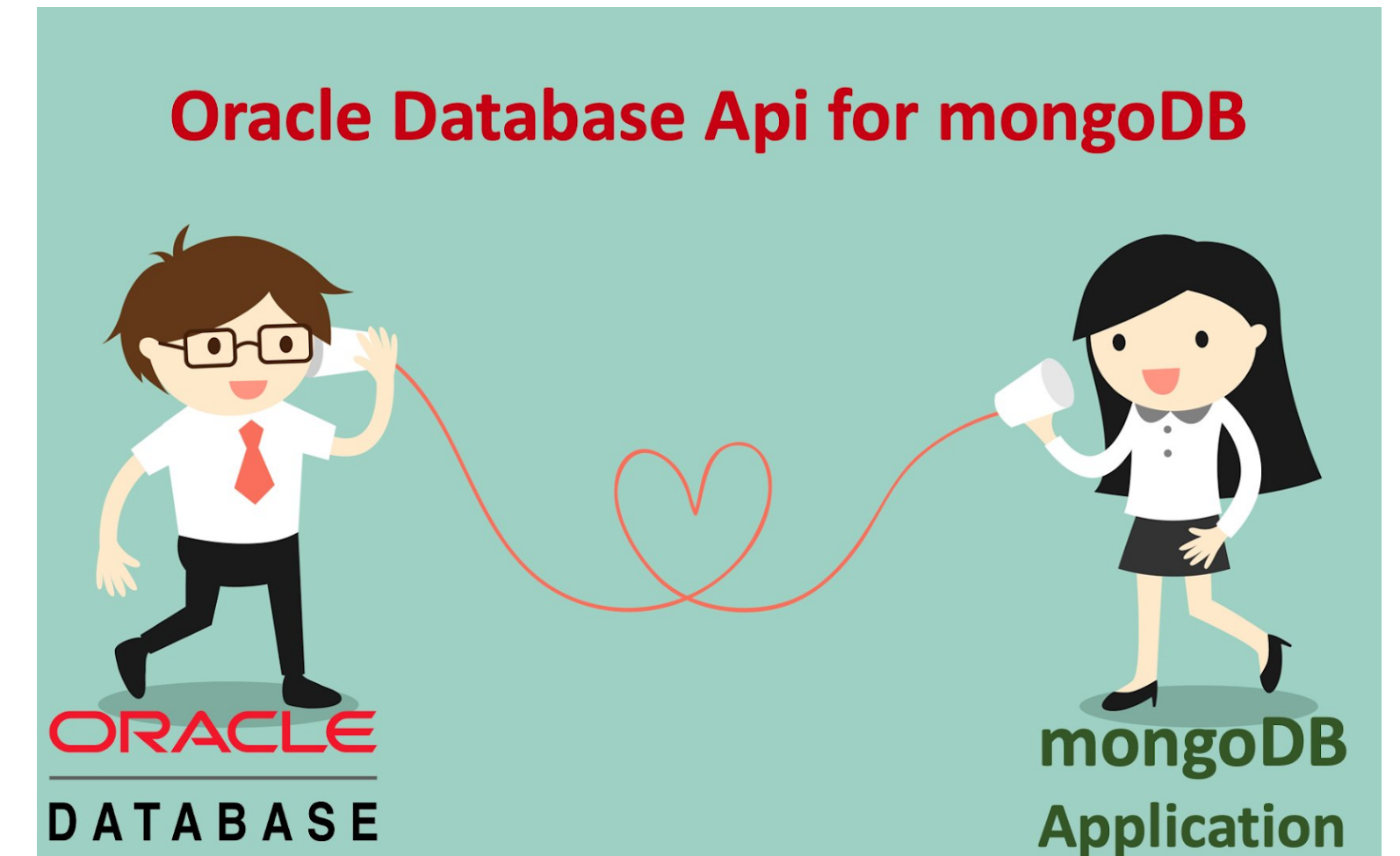
Alternatives?



**Amazon
DocumentDB**



**Azure
Cosmos DB**



**Informix database
software**

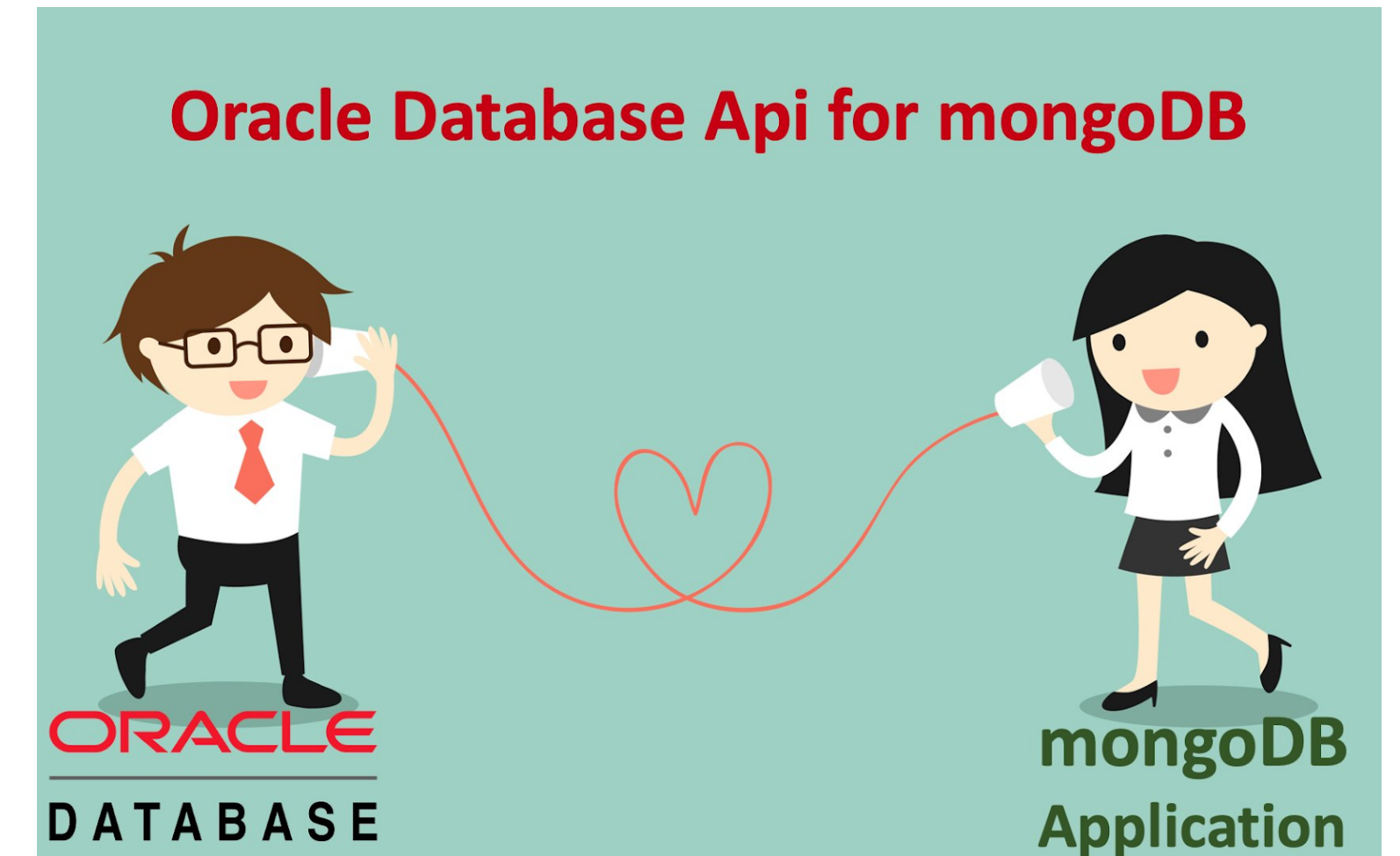
Alternatives?



**Amazon
DocumentDB**



**Azure
Cosmos DB**



IBM
Informix database
software

Our idea

Our idea

- Should work with unmodified MongoDB drivers and apps

Our idea

- Should work with unmodified MongoDB drivers and apps
- Should not store data itself

Our idea

- Should work with unmodified MongoDB drivers and apps
- Should not store data itself
- Should be a honest open-source solution

Who are you?

Who are you?



Peter Zaitsev, Advisor
Percona co-founder

Who are you?



Peter Zaitsev, Advisor
Percona co-founder



Peter Farkas, CEO
ex-Percona, Cloudera
10+ years of experience
leading database software
and services teams

Who are you?



Peter Zaitsev, Advisor
Percona co-founder



Peter Farkas, CEO
ex-Percona, Cloudera
10+ years of experience
leading database software
and services teams



Alexey Palazhchenko, CTO
ex-Percona
10+ years of experience leading
software engineering teams

FerretDB



FerretDB

- Stateless proxy for MongoDB wire protocol



FerretDB

- Stateless proxy for MongoDB wire protocol
- Stores data in PostgreSQL or Tigris



FerretDB

- Stateless proxy for MongoDB wire protocol
- Stores data in PostgreSQL or Tigris
- Apache License 2.0



FerretDB



FerretDB

- Launched one year ago



FerretDB

- Launched one year ago
- Released v0.7.0 several days ago



FerretDB

- Launched one year ago
- Released v0.7.0 several days ago
- v1.0.0 GA is planned for early February



FerretDB

- Launched one year ago
- Released v0.7.0 several days ago
- v1.0.0 GA is planned for early February
- Looking for early adopters (with small data sets)



Why PostgreSQL?

Why PostgreSQL?

- Open-source, huge community

Why PostgreSQL?

- Open-source, huge community
- Very reliable

Why PostgreSQL?

- Open-source, huge community
- Very reliable
- Good JSON support

Why PostgreSQL?

- Open-source, huge community
- Very reliable
- Good JSON support
- Many companies run both PostgreSQL and MongoDB

Storing BSON in jsonb

Storing BSON in jsonb

- Field order is significant in BSON documents

Storing BSON in jsonb

- Field order is significant in BSON documents
- Field order is not preserved by jsonb objects

Storing BSON in jsonb

- Field order is significant in {
BSON documents
 - Field order is not
preserved by jsonb
objects
- ```
 "$k": ["platform", "application"],
 "platform": "Node.js v14.17.3",
 "application": {
 "$k": ["name"],
 "name": "mongosh 1.0.1"
 }
 }
```

# Storing BSON in jsonb

# Storing BSON in jsonb

- Fields may start with «\$» in BSON documents

# Storing BSON in jsonb

- Fields may start with «\$» in BSON documents
  - \$getField, \$setField

# Storing BSON in jsonb

- Fields may start with «\$» in BSON documents
  - \$getField, \$setField
- Not supported

# Storing BSON in jsonb

# Storing BSON in jsonb

- BSON has more data types than JSON

# Storing BSON in jsonb

- BSON has more data types than JSON
  - int32, int64, double

# Storing BSON in jsonb

- BSON has more data types than JSON
  - int32, int64, double
  - timestamp

# Storing BSON in jsonb

- BSON has more data types than JSON
  - int32, int64, double
  - timestamp
  - JavaScript code with scope

# Storing BSON in jsonb

- BSON has more data types than JSON
  - int32, int64, double
  - timestamp
  - JavaScript code with scope
- Use JSON objects with special fields

# Storing BSON in jsonb

- BSON has more data types than JSON
  - int32, int64, double
  - timestamp
  - JavaScript code with scope
- Use JSON objects with special fields
  - 42, {«\$l»: 42}, {«\$f»: 42}

# Storing BSON in jsonb

- BSON has more data types than JSON
  - int32, int64, double
  - timestamp
  - JavaScript code with scope
- Use JSON objects with special fields
  - 42, {«\$l»: 42}, {«\$f»: 42}
  - {«\$t»: «42»}

# Storing BSON in jsonb

- BSON has more data types than JSON
  - int32, int64, double
  - timestamp
  - JavaScript code with scope
- Use JSON objects with special fields
  - 42, {«\$l»: 42}, {«\$f»: 42}
  - {«\$t»: «42»}
  - not supported

# Storing BSON in jsonb

# Storing BSON in jsonb

- Some BSON values are problematic

# Storing BSON in jsonb

- Some BSON values are problematic
  - null

# Storing BSON in jsonb

- Some BSON values are problematic
  - null
  - $\pm$ Infinity

# Storing BSON in jsonb

- Some BSON values are problematic
  - null
  - $\pm$ Infinity
  - NaN

# Storing BSON in jsonb

- Some BSON values are problematic
  - null
  - $\pm$ Infinity
  - NaN
  - negative zero

# Storing BSON in jsonb

- Some BSON values are problematic
  - null
  - $\pm$ Infinity
  - NaN
  - negative zero
- Use JSON values or objects with special fields

# Storing BSON in jsonb

- Some BSON values are problematic
  - null
  - $\pm$ Infinity
  - NaN
  - negative zero
- Use JSON values or objects with special fields
  - null

# Storing BSON in jsonb

- Some BSON values are problematic
  - null
  - $\pm$ Infinity
  - NaN
  - negative zero
- Use JSON values or objects with special fields
  - null
  - {«\$f»: «+Infinity»}

# Storing BSON in jsonb

- Some BSON values are problematic
  - null
  - $\pm$ Infinity
  - NaN
  - negative zero
- Use JSON values or objects with special fields
  - null
  - {«\$f»: «+Infinity»}
  - {«\$f»: «NaN»}

# Storing BSON in jsonb

- Some BSON values are problematic
  - null
  - $\pm$ Infinity
  - NaN
  - negative zero
- Use JSON values or objects with special fields
  - null
  - {«\$f»: «+Infinity»}
  - {«\$f»: «NaN»}
  - not actually stored

# Comparing BSON values

# Comparing BSON values

- We need to compare BSON values for filtering and sorting

# Comparing BSON values

- We need to compare BSON values for filtering and sorting
- Including values of different types

# Comparing BSON values

- We need to compare BSON values for filtering and sorting
- Including values of different types
- And problematic values

# Comparing BSON values

# Comparing BSON values

- NaN < 1

# Comparing BSON values

- NaN < 1
- null < NaN

# Comparing BSON values

- NaN < 1
- null < NaN
- [] < null

# Comparing BSON values

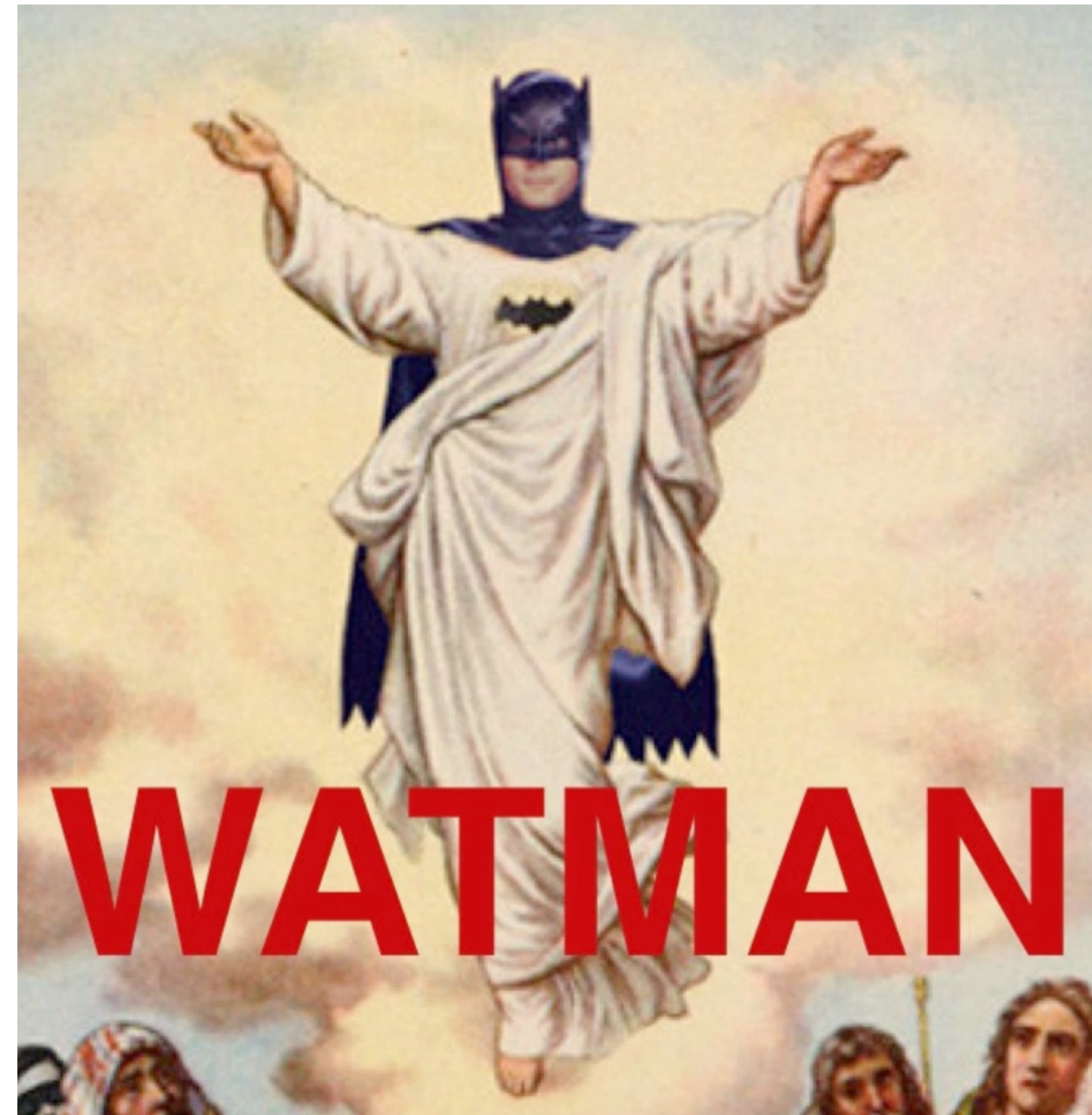
- $\text{NaN} < 1$
- $\text{null} < \text{NaN}$
- $[] < \text{null}$
- $[] < [\text{null}]$

# Comparing BSON values

- $\text{NaN} < 1$
- $\text{null} < \text{NaN}$
- $[] < \text{null}$
- $[] < [\text{null}]$
- $\text{null} < /> [\text{null}]$   
(depends on sorting order)

# Comparing BSON values

- $\text{NaN} < 1$
- $\text{null} < \text{NaN}$
- $[] < \text{null}$
- $[] < [\text{null}]$
- $\text{null} < /> [\text{null}]$   
(depends on sorting order)



<https://www.destroyallsoftware.com/talks/wat>

# Comparing BSON values

# Comparing BSON values

- Do a comparison in Go

# Comparing BSON values

- Do a comparison in Go
- Fetch all data

# Comparing BSON values

- Do a comparison in Go
- Fetch all data
- Add great integration tests

# Comparing BSON values

- Do a comparison in Go
- Fetch all data
- Add great integration tests
- Remove support for edge-cases

# Comparing BSON values

- Do a comparison in Go
- Fetch all data
- Add great integration tests
- Remove support for edge-cases
- Pushdown simple queries

# Comparing BSON values

- Do a comparison in Go
- Fetch all data
- Add great integration tests
- Remove support for edge-cases
- Pushdown simple queries
- Translate Go code to PL/pgSQL?

# Comparing BSON values

- Do a comparison in Go
- Fetch all data
- Add great integration tests
- Remove support for edge-cases
- Pushdown simple queries
- Translate Go code to PL/pgSQL?
- Add custom operator?

# Comparing BSON values

- Do a comparison in Go
- Fetch all data
- Add great integration tests
- Remove support for edge-cases
- Pushdown simple queries
- Translate Go code to PL/pgSQL?
- Add custom operator?
- Add custom data type?

# Planned changes

# Planned changes

- Remove support for problematic values

# Planned changes

- Remove support for problematic values
- Separate types and values

# Planned changes

- Remove support for problematic values
- Separate types and values
  - $\{\llbracket v \rrbracket : \{\llbracket \$l \rrbracket : 42\}\}$

# Planned changes

- Remove support for problematic values
- Separate types and values
  - {«v»: {«\$l»: 42}}
  - {«\$types»: {«v»: «int64»}, «v»: 42}

# Planned changes

- Remove support for problematic values
- Separate types and values
  - {«v»: {«\$l»: 42}}
  - {«\$types»: {«v»: «int64»}, «v»: 42}
- Make jsonb indexes work

# Planned changes

- Remove support for problematic values
- Separate types and values
  - {«v»: {«\$l»: 42}}
  - {«\$types»: {«v»: «int64»}, «v»: 42}
- Make jsonb indexes work
  - WHERE \_jsonb->'v' = 42

# Integration tests

# Integration tests

- Our own compatibility tests

# Integration tests

- Our own compatibility tests
- Tests of existing applications

# Integration tests

- Our own compatibility tests
- Tests of existing applications
- «MongoDB API Tester» tests

# Integration tests

- Our own compatibility tests
- Tests of existing applications
- «MongoDB API Tester» tests
- Community testers

# Community

# Community

- Roadmap:

<https://github.com/orgs/FerretDB/projects/2>

# Community

- Roadmap:

<https://github.com/orgs/FerretDB/projects/2>

- Please star us now:

<https://github.com/FerretDB/FerretDB>

# Community

- Roadmap:  
<https://github.com/orgs/FerretDB/projects/2>
- Please star us now:  
<https://github.com/FerretDB/FerretDB>
- And give it a try tomorrow

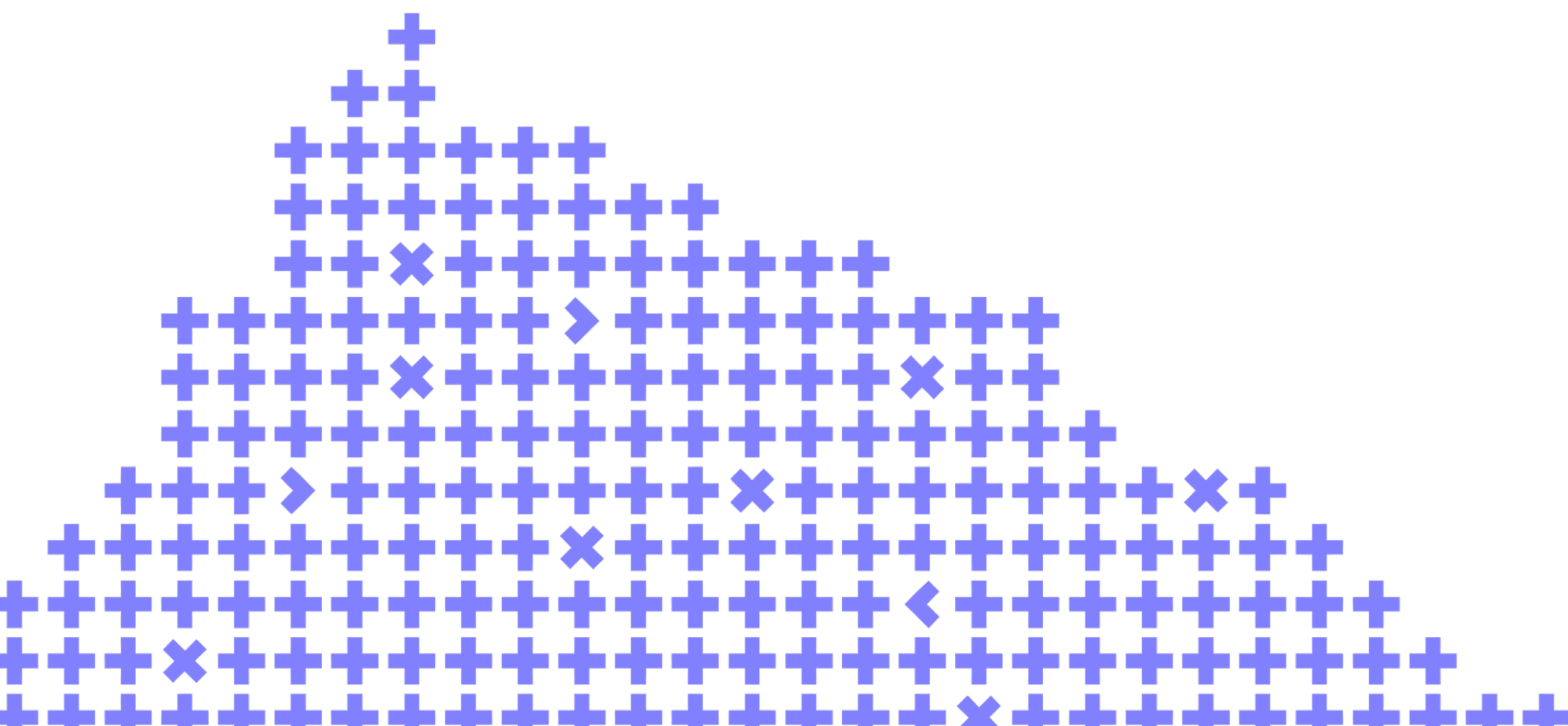
Thank you!  
Questions?

<https://ferretdb.io/>

<https://github.com/FerretDB>

[https://twitter.com/ferret\\_db](https://twitter.com/ferret_db)

<https://techhub.social/@ferretdb>



**Leave your feedback!**



Co-organizer

**Yandex**